

Decomposition of Human Motion into Dynamics Based Primitives with Application to Drawing Tasks [★]

Domitilla Del Vecchio ^a, Richard M. Murray ^a, Pietro Perona ^a

^a *California Institute of Technology, 1200 E. California Blvd, Pasadena 91125 CA*

Abstract

Using tools from dynamical systems and systems identification we develop a framework for the study of primitives for human motion, which we refer to as *movemes*. The objective is understanding human motion by decomposing it into a sequence of elementary building blocks that belong to a known alphabet of dynamical systems. We develop a segmentation and classification algorithm in order to reduce a complex activity into the sequence of movemes that have generated it. We test our ideas on data sampled from five human subjects who were drawing figures using a computer mouse. Our experiments show that we are able to distinguish between movemes and recognize them even when they take place in activities containing an unspecified number of movemes.

Key words: Classification; parameter estimation; learning theory; data acquisition; computer experiments; signal segmentation.

1 Introduction

Building systems that can detect and recognize human actions and activities is an important goal of modern engineering. Applications range from human-machine interfaces to security to entertainment. With the development of information technology we can expect that computer systems will be increasingly embedded in our environment, so that human-machine interaction will need interfaces that are easier to use and more natural. As humans use their visual system and auditory system to communicate, several works (see for example [12,22] and the earlier work on building human-machine interfaces using vision [9,16,25,26,23]) ask the question of whether it is possible to develop computerized equipment able to communicate with humans in similar way. As described extensively in [4] there is also an immediate need for automated surveillance systems in commercial, law enforcement, and military applications.

A fundamental problem in detecting and recognizing human action is one of representation. Our point of view is

that human activity should be decomposed into building blocks which belong to an “alphabet” of elementary actions; for example the activity “answering the phone” could be decomposed into the sequence “step-step-step-reach-lift”, where “step”, “reach” and “lift” may not be further decomposed. We refer to these primitives of motion as *movemes*. Our aim is then to build an alphabet of movemes, which one can compose to represent and describe human motion similar to the way phonemes are used in speech. The word “moveme” intended as primitive of motion was invented by [3]. They studied periodic or stereotypical motions such as walking or running where the motion is always the same and therefore their movemes, like the phonemes, were repeatable segments of trajectory. [8] studied motions that were parametrized by an initial condition and a target, such as “reach” that requires the specification of a target location. They proposed that movemes ought to be parametrized by goal and style parameters. Their moveme models are phenomenological and non-causal. In this paper we attempt to define movemes in terms of causal dynamical systems. This approach opens the possibility of dealing with problems like prediction, and leads to more compact models parameterized by a small number of parameters. Moreover the dynamical systems framework allows us to use a set of mathematical tools for determining analytically the performance of the algorithms proposed.

The idea of dynamical primitives of motion has also ap-

[★] This paper was not presented at any IFAC meeting. Corresponding author D. Del Vecchio. Tel. +1 (626)3952289. Fax +1 (626) 7968914.

Email addresses: ddomitilla@cds.caltech.edu (Domitilla Del Vecchio), murray@cds.caltech.edu (Richard M. Murray), perona@caltech.edu (Pietro Perona).

peared in neurobiology studies. [2] pose the question whether the motor behavior of vertebrates is based on simple units (motor primitives) that can be combined flexibly to accomplish a variety of motor tasks, and experiments have provided evidence for a modular organization of the spinal cord in frogs and rats. [17] ran experiments which showed that the fields induced by the focal activation of the spinal cord follow a principle of vectorial summation, so that a variety of motor control policies can be obtained from a simple linear combination of few control modules. Experimental results in [11] and [7] support the idea that kinematic and dynamic internal models are utilized in movement planning and control. The “internal model” hypothesis proposes that the brain acquires an inverse dynamic model of the object to be controlled through motor learning after which motor control can be executed mostly in a feed-forward manner. Thus, the role of dynamics in the description of human motion seems to be an important one.

What is the alphabet of movemes? Which are the dynamical models that we should use to represent them? Can a continuous trajectory of a human body be decomposed automatically into its component movemes? To answer these questions we take a relatively abstract point of view so to find a representation framework that may apply to situations where dynamical evolution and switching between different dynamical modes come into play. We introduce a formal definition of a moveme and set up the classification and segmentation problem that can be appropriately formalized in a dynamical systems framework. Standard system identification tools and stability arguments can then be applied to derive analytical error analysis for the proposed algorithm so as to obtain performance estimates in the presence of noise and modeling uncertainties. Finally we present some experimental results on human drawing data. Even though the particular example considered can be solved other ways, it is meant to show how the developed techniques can be used in a practical and simple application characterized by modeling uncertainty, noise, and subject variability.

The problem of segmenting data streams originating from different unknown or partially known processes which alternate in time is a general problem of interest to various areas, see for example [10,13,24]. We propose a solution to the problem in our particular scenario in which each one of the segments has been generated from the perturbed version of a linear dynamical system belonging to a finite known set of possible linear models. By using system identification techniques [14,20] and pattern recognition techniques [1,21] we develop an off-line joint segmentation and classification algorithm and provide analytical error analysis. The dynamical systems representation for describing human motion is not a novel idea, some sample citations include [19,15,18]. Our contribution lies mainly in the development of a joint classification-segmentation algorithm, based on *a priori* given classes of motion (the moveme alphabet),

and characterized by a detailed error analysis.

This paper is organized as follows. In Section 2 we define *movemes* formally according to a dynamical systems framework and we introduce the classification problem. In Section 3 we introduce the joint segmentation and classification problem, and in Section 4 propose a solution. In Section 5, we test our ideas on data sampled from five human subjects engaged in drawing houses, cars, ships and suns using a computer mouse.

2 Dynamical Definition of Moveme

We provide in this section some theoretical background, a formal definition of moveme in the dynamical systems framework, and introduce the classification problem.

2.1 Definitions, properties, choice of model class

Let $M(\Theta)$ denote a linear time invariant (LTI) system class parameterized by $\Theta \in E$, E a linear space, and let \mathcal{U} denote a class of inputs. Let $y(t) = Y(M(\Theta)|_{u,x_0})(t)$, for $t \geq t_0$, denote the output of $M(\Theta)$ once parameter $\Theta \in E$, input $u \in \mathcal{U}$, and initial conditions x_0 have been chosen. Let $\theta \in E' \subset E$ be a parameter lying in a subspace of E , and define a map $\Upsilon : E \rightarrow E'$. We write $\theta = \Upsilon(\Theta)$ to represent the transformation from $\Theta \in E$ to the reduced set of parameters $\theta \in E'$.

Definition 1 Let $M^1 = \{M(\Theta)|\theta \in \mathcal{C}^1\}$ and $M^2 = \{M(\Theta)|\theta \in \mathcal{C}^2\}$ denote two subsets in M with $\mathcal{C}^j \subset E'$ for $j = 1, 2$. M^1 and M^2 are said to be *dynamically independent* if

(i) the class of systems M and the class of inputs \mathcal{U} are such that

$$Y(M(\Theta_1)|_{u_1,x_0})(t) = Y(M(\Theta_2)|_{u_2,x_0})(t), \quad \forall t \geq t_0$$

if and only if $(\Theta_1, u_1) = (\Theta_2, u_2)$ for $u_1 \in \mathcal{U}$ and $u_2 \in \mathcal{U}$;

(ii) the sets \mathcal{C}^1 and \mathcal{C}^2 are non empty, bounded, and have trivial intersection, i.e. $\mathcal{C}^1 \cap \mathcal{C}^2 = \{\emptyset\}$.

Each of the elements of a set \mathcal{M} of mutually dynamically independent model sets is called a *moveme*. Note that, in case $u_1 = u_2$, property (i) is equivalent to the identifiability property of class M ([14]).

In this paper, we choose our model class M and input u as asymptotically stable linear systems driven by a unit step input with full state output, that is

$$\begin{aligned} \dot{x} &= Ax + b \\ y &= x, \end{aligned} \tag{1}$$

where $A \in \mathbb{R}^{n \times n}$, $x = (x_1, \dots, x_n) \in \mathbb{R}^n$, $b \in \mathbb{R}^n$, so that $\Theta = (A|b) \in E = \mathbb{R}^{n \times (n+1)}$ and $\theta = A \in E' = \mathbb{R}^{n \times n}$, with $\Upsilon(A|b) = A$. Given any signal $x(t)$ we can determine a good representative of such a signal in the class of models (1) by minimizing the cost function (see for example [14]):

$$(\hat{A}|\hat{b}) = \arg \min_{(A|b)} \frac{1}{2} \int_{t_0}^T (\dot{x} - (A|b)\bar{x})^T (\dot{x} - (A|b)\bar{x}) dt \quad (2)$$

with $\bar{x} = (x^T, 1)^T$, which gives the least squares estimate of parameters $(\hat{A}|\hat{b})$ so to get the estimate of x in model class (1) as $\hat{x} = \hat{A}\hat{x} + \hat{b}$, with $\hat{x}(t_0) = x(t_0)$. In this paper we take a relative straightforward approach to estimation, more sophisticated methods exist for estimation of parameters in presence of noise. We leave this to future work.

For the class of models (1) we make the following assumption.

Assumption 2.1 Given $x(t)$ as the output of model (1) we assume that the initial condition x_0 is such that for any $v \in \mathbb{R}^{n+1}$, $v^T \bar{x}(t) = 0$, $t \in [t_1, t_2]$, $t_2 > t_1$ implies $v = 0$, where $\bar{x} = (x^T, 1)^T$.

This assumption means that $x(t)$ cannot also be described by a lower order dynamical system. In fact if $v^T \bar{x}(t) = 0$, $t \in [t_1, t_2]$, $t_2 > t_1$ for some $v \neq 0$ then $x_n(t) = \alpha_0 + \alpha_1 x_i(t) + \dots + \alpha_{n-1} x_{n-1}(t)$ for any t , therefore the dynamics can be described just in terms of $x_1(t), \dots, x_{n-1}(t)$ and $x_n(t)$ can be derived algebraically. A direct consequence of such an assumption is given by the following lemma.

Lemma 2 Let $x(t)$ and $z(t)$ be generated by two LTI systems $\dot{x} = A_1 x + b_1$, and $\dot{z} = A_2 z + b_2$, and let Assumption 2.1 hold. Then $z(t) = x(t)$ for all t if and only if $(A_1|b_1) = (A_2|b_2)$.

Proof (\Leftarrow) If $(A_1|b_1) = (A_2|b_2)$ then $z(t) = x(t)$ for all t by uniqueness of solutions. (\Rightarrow) If $z(t) = x(t)$ for all t then $\dot{z}(t) = \dot{x}(t)$ for all t , so that $A_1 x + b_1 = A_2 z + b_2$. This implies $[(A_1|b_1) - (A_2|b_2)] \bar{x}(t) = 0$ for all t , which by Assumption 2.1 (applied to each row) implies $(A_1|b_1) = (A_2|b_2)$. \square

This lemma shows that property (i) of Definition 1 is satisfied by our choice of M and \mathcal{U} . Property (ii) is verified by choosing for example \mathcal{C}^j , $j = 1, \dots, m$, as balls in $\mathbb{R}^{n \times n}$ with centers $A_c^j \in \mathbb{R}^{n \times n}$, $j = 1, \dots, m$, and radii r_j , such that:

$$\begin{aligned} \mathcal{C}^j &= B_{r_j}(A_c^j), & j &= 1, \dots, m \\ \mathcal{C}^j \cap \mathcal{C}^k &= \{\emptyset\}, & j &\neq k \end{aligned} \quad (3)$$

where m is the number of movemes and the matrix norm is the Frobenius norm. In what follows we assume that the sets \mathcal{C}^j are described by equation (3). Then we have constructed a set $\mathcal{M} = \{M^1, \dots, M^m\}$ of m movemes where $M^k = \{M((A|b)) | A \in \mathcal{C}^k\}$, for $k \in \{1, \dots, m\}$ and M is in the form given by equation (1).

2.2 Classification Problem

Let the signal $x(t)$ be generated by the perturbed version of (1):

$$\begin{aligned} \dot{x} &= Ax + b + d(t) \\ y &= x \end{aligned} \quad (4)$$

with $d(t)$ a bounded realization of white noise. We assume that $A \in \mathcal{C}^j$, for some $j \in \{1, \dots, m\}$, and we will say that the signal $x(t)$ is of class j . Under what conditions on A and $d(t)$ can we classify $x(t)$ correctly? Since $A \in \mathcal{C}^j$, there exists $\delta < r_j$ such that $A = A_c^j + \delta U$ with U a unit norm matrix and A_c^j center of \mathcal{C}^j . Then the problem of classifying $x(t)$ is to find conditions on δ and $d(t)$, such that the estimate \hat{A} of A obtained by equation (2) is lying in \mathcal{C}^j . If $d(t) = 0$ we can exactly identify A and then correctly classify $x(t)$. The presence of $d(t)$ induces an estimation error so that \hat{A} will not be equal to A , but it is not necessary to achieve equality for our purpose as the following lemma shows.

Lemma 3 Let $x(t)$, $t \in [t_0, T]$ be generated by system (4), where $A = A_c^j + \delta U$, for U unit norm matrix, and δ a positive constant. Let A_c^j be defined by relations (3). Let \hat{A} be the estimate of A according to equation (2). There exist positive constants $\bar{\delta}$ and \bar{d} such that if $\delta \leq \bar{\delta}$ and $\|d(t)\| \leq \bar{d}$, then $\arg_{k \in \{1, \dots, j, \dots, m\}} \{\|\hat{A} - A_c^k\| \leq r_k\} = j$.

Proof (sketch) By equation (2) we have

$$(\hat{A}|\hat{b}) = \left(\int_{t_0}^T \dot{x}(t) \bar{x}(t)^T dt \right) \left(\int_{t_0}^T \bar{x}(t) \bar{x}(t)^T dt \right)^{-1} \quad (5)$$

where $\left(\int_{t_0}^T \bar{x}(t) \bar{x}(t)^T dt \right)^{-1}$ is well defined if either $d(t) = 0$ by Assumption 2.1, or $d(t) \neq 0$ by the fact that $d(t)$ is realization of white noise that is uncorrelated in time. Then we substitute in (5) the expression of \dot{x} given in equation (4), with $A = A_c^j + \delta U$, we subtract by both sides $(A_c^j|b)$, and we take the norm to get $\|\hat{A} - A_c^j\| \leq \bar{\delta} + \bar{d} c$, for a suitable constant c . Therefore it is sufficient that $\bar{\delta} + \bar{d} c \leq r_j$, which is verified for suitable $\bar{\delta}$ and \bar{d} . \square

In this section we have highlighted that the basic requirement for solving the classification problem is the one of

having non intersecting sets in parameter space characterizing the sets of dynamical models M^j , $j = 1, \dots, m$. In practice the moveme sets are derived from data, i.e. signals sets that we think define movemes, and not necessarily the parameter sets representing them are not intersecting. This may be due to noise, or to the fact that two candidate movemes are in fact not defining a set of two movemes. In this second case we say that the signals sets we analyzed are not *well-posed* according to the chosen model class M . The well-posedness property can be hard to establish, and a more rigorous treatment can be found in [5,6].

3 Problem Statement

Consider the sequence of systems for $i = 0, \dots, l$

$$\begin{cases} \dot{x} = (A_i + \delta U_i)x + b_i + d(t) & t \in [\tau_{i-1}, \tau_i) \\ \dot{x} = (A_{i+1} + \delta U_{i+1})x + b_{i+1} + d(t) & t \in [\tau_i, \tau_{i+1}) \end{cases} \quad (6)$$

with $x \in \mathbb{R}^n$, $A_i \in \mathbb{R}^{n \times n}$, $b_i \in \mathbb{R}^n$, $U_i \in \mathbb{R}^{n \times n}$ norm one matrices (according to Frobenius norm), $\delta \in \mathbb{R}$ modeling uncertainty, $d(t)$ realization of white noise. We make the following assumptions:

Assumption 3.1 $A_i \in \mathbb{R}^{n \times n}$ an unknown matrix whose value can take place in the set of known Hurwitz matrices $\{A_c^1, \dots, A_c^m\}$, which are centers of the sets defined in (3), i.e. $\mathcal{C}^j = B_{r_j}(A_c^j)$ with $\mathcal{C}^j \cap \mathcal{C}^k = \{\emptyset\}$ for $j \neq k$. Modeling uncertainty δ is such that $(A_i + \delta U_i) \in \mathcal{C}^j$ for some j , that is segment i is of class j .

Assumption 3.2 Vectors $b_i \in \mathbb{R}^n$ are unknown constant vectors, τ_i unknown switching times with τ_0 known starting time and τ_l known ending time. The total number of switching times l is unknown.

Assumption 3.3 Modeling uncertainty and disturbance are bounded, i.e. $|\delta| \leq \bar{\delta}$, and $\|d(t)\| \leq \bar{d}$.

Assumption 3.4 The nominal system obtained letting in system (6) noise and parameter uncertainty to zero

$$\begin{cases} \dot{x} = A_i x + b_i & t \in [\tau_{i-1}, \tau_i) \\ \dot{x} = A_{i+1} x + b_{i+1} & t \in (\tau_i, \tau_{i+1}] \end{cases} \quad (7)$$

satisfies the *interconnection condition*

$$\frac{\dot{x}(\tau_i^-)^T \dot{x}(\tau_i^+)}{\|\dot{x}(\tau_i^-)\| \|\dot{x}(\tau_i^+)\|} \leq \rho_0 < 1. \quad (8)$$

where we define $\dot{x}(\tau_i^-) = \lim_{\tau \rightarrow \tau_i^-} \dot{x}(\tau)$ and $\dot{x}(\tau_i^+) = \lim_{\tau \rightarrow \tau_i^+} \dot{x}(\tau)$. It also satisfies Assumption 2.1.

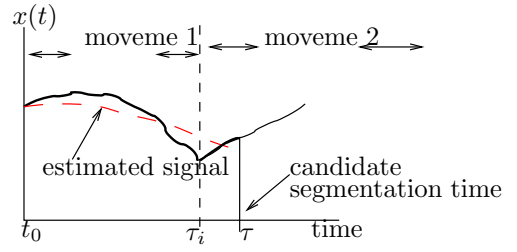


Fig. 1. Signal considered for computation of approximation and parametric errors (bold line) and estimated signal \hat{x} (dashed line.)

The interconnection condition gives a bound on the discontinuity in the trajectory's derivative at the switching points.

We wish to obtain sufficient conditions on noise level and parameter uncertainty that allow off-line determination of the sequence of times $\{\tau_1, \dots, \tau_{l-1}\}$ and the sequence of matrices $\{A_1, \dots, A_{l-1}\}$ from the observation of state x . If we have a good guess of the switching times, then we can apply Lemma 3 so to solve the classification problem in each interval between two switching times.

We thus focus our attention on the segmentation part of the problem, that is to determine a good estimate for the sequence of times $\{\tau_1, \dots, \tau_{l-1}\}$. We use an iterative approach in which at each iteration we look for the maximizer of a function defined on $[t_0, t_M]$ where $t_M = \tau_l$ and t_0 is a starting time which coincides with τ_0 at the first iteration. We want to show the maximizer of such function falls in an interval I around the first switching time encountered after time t_0 ; moreover this interval should shrink down to the switching point when noise and parameter uncertainty go to zero. To define such a function we define three quantities for system (6): the transition factor at time τ , the approximation error, and the parametric error. The transition factor is defined as

$$\begin{aligned} \text{Tr}(\tau) &= \frac{1}{2} \left(1 - \frac{\dot{x}_{av}(\tau^-)^T \dot{x}_{av}(\tau^+)}{\|\dot{x}_{av}(\tau^-)\| \|\dot{x}_{av}(\tau^+)\|} \right) \\ \dot{x}_{av}(\tau^-) &:= \frac{1}{\Delta\tau} \int_{\tau-\Delta\tau}^{\tau} \dot{x}(t) dt \\ \dot{x}_{av}(\tau^+) &:= \frac{1}{\Delta\tau} \int_{\tau}^{\tau+\Delta\tau} \dot{x}(t) dt, \end{aligned} \quad (9)$$

which takes care of local properties of the signal $x(t)$. $\Delta\tau$ is a positive constant depending on perturbation level which will be determined later. The approximation and the parametric errors are computed based on the observation of $x(t)$ of system (6) for $t \in [t_0, \tau]$, $\tau \in (t_0, t_M)$. We begin by computing the estimates of A and b according to (2). Assuming that $t_0 = \tau_{i-1}$, and letting

$\bar{x} := (x^T, 1)^T$, we have:

$$(\hat{A}|\hat{b})(\tau, t_0) = \left[\int_{t_0}^{\tau_i} (A_i + \delta U_i |b_i) \bar{x} \bar{x}^T dt + \int_{\tau_i}^{\tau} (A_{i+1} + \delta U_{i+1} |b_{i+1}) \bar{x} \bar{x}^T dt + \int_{t_0}^{\tau} d(t) \bar{x} \bar{x}^T dt \right] \left[\int_{t_0}^{\tau} \bar{x} \bar{x}^T dt \right]^{-1} \quad (10)$$

for $\tau_i < \tau < \tau_{i+1}$, and

$$(\hat{A}|\hat{b})(\tau, t_0) = \left[\int_{t_0}^{\tau} (A_i + \delta U_i |b_i) \bar{x} \bar{x}^T dt + \int_{t_0}^{\tau} d(t) \bar{x} \bar{x}^T dt \right] \left[\int_{t_0}^{\tau} \bar{x} \bar{x}^T dt \right]^{-1} \quad (11)$$

for $\tau \leq \tau_i$. The estimates given by (10) and (11) generate the system

$$\dot{\hat{x}} = \hat{A}(\tau, t_0) \hat{x} + \hat{b}(\tau, t_0), \quad \hat{x}(t_0) = x(0). \quad (12)$$

This situation is depicted in Figure 1, where we report the candidate segmentation time τ , the switching time τ_i , the portion of signal under study composed by the sequence of two movemes (bold line), the estimated trajectory (dashed line) obtained by system (12). We define the parametric error at time τ as

$$e_p(\tau, t_0) = \min_{j=1, \dots, m} \|\hat{A}(\tau, t_0) - A_c^j\|, \quad (13)$$

where we consider the Frobenius norm to be the matrix norm. The approximation error at time τ is

$$e_a(\tau, t_0) = \frac{1}{\tau - t_0} \int_{t_0}^{\tau} (x - \hat{x})^T (x - \hat{x}) dt, \quad (14)$$

For each time τ we have different values of these three quantities and the idea is to combine them in one function of τ which has the maximizer close to the switching point. The function we choose is

$$W(\tau, t_0) = \frac{\exp(\frac{-e_p(\tau, t_0)^2}{\sigma^2}) \text{Tr}(\tau)}{a + e_a(\tau, t_0)}, \quad \tau \in (t_0, t_M] \quad (15)$$

where a is an arbitrarily small positive constant to prevent the denominator from being zero. By maximizing function $W(\tau)$ we are looking for the value of τ which has small approximation error (which implies a good guess of dynamical parameters), small parametric error, and a high transition factor. Expression (9) involves integration over time $\Delta\tau$ to attenuate the effect of noise and its expression for system (7) is obtained by letting $\Delta\tau \rightarrow 0$. In this case we find that $\text{Tr}(\tau_i) \geq (1 - \rho_0)/2$

and for $\tau \neq \tau_i$, $\text{Tr}(\tau) = 0$. The idea of the transition factor term is to preserve this property as much as possible in the perturbed case so that all the times $\tau_i + \Delta\tau \leq \tau \leq \tau_{i+1} - \Delta\tau$ or $t_0 < \tau \leq \tau_i - \Delta\tau$ are penalized with respect to time τ_i . We also choose to minimize $e_p(\tau, t_0)$ so to reduce the effect of perturbation on the parameter estimates. Alternatively, one could constrain the estimates \hat{A} to lie in a ball around A_i , but we do not know the value of A_i *a priori*, we just know that it belongs to a set of possible values. Therefore we decide to minimize the distance of \hat{A} from the closest point A_j at time τ according to a Gaussian metric.

4 Main Result

We give the following theorem.

Theorem 4 *Consider the sequence of dynamical systems given in (6) subject to Assumptions 3.1- 3.4. Let the function $W(\tau, t_0)$ be defined as*

$$W(\tau, t_0) = \frac{\exp(\frac{-e_p(\tau, t_0)^2}{\sigma^2}) \text{Tr}(\tau)}{a + e_a(\tau, t_0)}, \quad \tau \in (t_0, t_M]$$

for $t_0 = \tau_{i-1}$ and $t_M = \tau_i$. Then there exist bounds δ^* and d^* such that if $\bar{\delta} \leq \delta^*$ and $\bar{d} \leq d^*$ the potential function $W(\tau)$ admits its global maximizer $\hat{\tau}_i$ for $\hat{\tau}_i \in I = [\tau_i - \Delta\tau, \tau_i + \Delta\tau]$ where I contracts to τ_i as $\bar{\delta} \rightarrow 0$ and $\bar{d} \rightarrow 0$. Moreover the estimated class \hat{j} of the segment in $[t_0, \hat{\tau}_i]$ is j .

In what follows we omit the dependence on t_0 .

4.1 Intermediate results and proof of the theorem

To prove Theorem 4 we make use of a sequence of lemmas, whose proofs are sketched here. For detailed proofs see [5].

Lemma 5 *Consider expressions (10) and (11), with $\delta = 0$ and $d(t) = 0$ for all t , for system (7). There exists $k_1 > 0$ such that $\|(\hat{A}|\hat{b}) - (A_i|b_i)\|^2 \geq k_1(\tau - \tau_i)^2$ for $\tau_i < \tau < \tau_{i+1}$.*

Proof (sketch) Compute expression (10), and subtract $\Theta_i := (A_i|b_i)$ in both sides to get

$$\tilde{\Theta}(\tau) = \left[\int_{\tau_i}^{\tau} (\Theta_{i+1} - \Theta_i) \bar{x} \bar{x}^T dt \right] \left[\int_{t_0}^{\tau} \bar{x} \bar{x}^T dt \right]^{-1}, \quad (16)$$

where $\tilde{\Theta}(\tau) = (\hat{A}|\hat{b}) - (A_i|b_i)$. Define $g(\tau) = \|\tilde{\Theta}(\tau)\|^2 = \text{tr}(\tilde{\Theta}(\tau)^T \tilde{\Theta}(\tau))$, and apply Taylor's theorem at $\tau = \tau_i$. \square

This lemma establishes that the closer τ is to the switching time τ_i the smaller the lower bound of the parameter estimation error for the nominal system.

Lemma 6 Consider system (7) with estimates (10) and (11) with $\delta = 0$ and $d(t) = 0$ for all t . For the estimated states generated by system (12) and $e_a(\tau)$ as defined in (14), there exists $k_2 > 0$ such that $e_a(\tau) \geq k_2 \|(\hat{A}|\hat{b}) - (A_i|b_i)\|^2$, for $\tau_i < \tau < \tau_{i+1}$.

Proof (sketch) Consider the expression for the approximation error given in (14). Let $\tilde{x} = x - \hat{x}$, and note that the dynamics of \tilde{x} can be written as $\dot{\tilde{x}} = A\tilde{x} + \tilde{\Theta}\tilde{x}$, where $\tilde{\Theta} = (A|b) - (\hat{A}|\hat{b})$, and $(A|b)$ can be either $(A_i|b_i)$ either $(A_{i+1}|b_{i+1})$. The variable \hat{x} is the state of (12) with the parameter estimates obtained from (10) and (11) with $d(t) = 0$ and $\delta = 0$, $\tilde{x} = (\hat{x}^T, 1)^T$. Then we can review \tilde{x} as a function of $\tilde{\Theta}$, so that also e_a given in (14) can be reviewed as a function of $\tilde{\Theta}$. Let $g(\tilde{\Theta}) = e_a(\tilde{\Theta})$, and apply the Taylor's theorem to the function $g(\tilde{\Theta})$ at $\tilde{\Theta} = 0$. \square

This lemma states that for the nominal system (7) the approximation error lower bound increases as the parameter estimates become far from the parameters A_i, b_i . These two lemmas hold for the quantities computed for the nominal system. In order to link such quantities with the ones computed for the perturbed system (6), we give two additional results. The first one establishes how far two system's states are from each other when the two systems differ because of parameter differences and the presence of noise. The second result explicitly links parameter and approximation errors for nominal and perturbed systems through the aid of the first result.

Lemma 7 Let A and A_1 be Hurwitz matrices and consider the pair of systems $\dot{x} = Ax + b$ and $\dot{z} = A_1z + b_1 + d(t)$, with x and z in \mathbb{R}^n , $A, A_1 \in \mathbb{R}^{n \times n}$, b and b_1 constant vectors in \mathbb{R}^n , $\|d(t)\| \leq \bar{d}$ and $\|(A|b) - (A_1|b_1)\| \leq \bar{\delta}$. Then if $x(0) = z(0)$ there exist $k_3 > 0$ and $k_4 > 0$ such that $\|x - z\| \leq k_3\bar{\delta} + k_4\bar{d}$ for all $t \geq 0$.

Proof (sketch) Let $e = x - z$, and construct the error system. Then consider the Lyapunov function $V = e^T P e$ with $P > 0$ such that $PA_1 + A_1^T P + Q = 0$ for some $Q > 0$, and compute its derivative. \square

Lemma 8 Let $e_p(\tau)$ and $e_a(\tau)$ denote parametric errors and approximation errors given in expressions (13) and (14) for the sequence of dynamical systems (6). Let $e_p^0(\tau)$ and $e_a^0(\tau)$ denote parametric errors and approximation errors for the related nominal system (7). Then there

exist constants $k_p > 0$ and $k_a > 0$ such that

$$e_p^0(\tau) - \Delta \leq e_p(\tau) \leq e_p^0(\tau) + \Delta \quad (17)$$

$$e_a^0(\tau) - \varepsilon \leq e_a(\tau) \leq e_a^0(\tau) + \varepsilon \quad (18)$$

with $\Delta = k_p(\bar{d} + \bar{d}^2 + \bar{d}^3 + \bar{\delta} + \bar{\delta}^2 + \bar{\delta}^3)$ and $\varepsilon = k_a(\bar{d} + \bar{d}^2 + \bar{d}^3 + \bar{\delta} + \bar{\delta}^2 + \bar{\delta}^3 + \bar{\delta}^4 + \bar{\delta}^6)$.

Proof (sketch) . Let us first prove relation (17). Substitute in expression (10) and (11) $x = x^0 + \tilde{x}$ where x^0 is the state of the nominal system (7), and from Lemma (7) $\|\tilde{x}\| \leq k_1\bar{\delta} + k_2\bar{d}$. Let $\hat{\Theta} = (\hat{A}|\hat{b})$, and let $\hat{\Theta}^0 = (\hat{A}^0|\hat{b}^0)$ be the estimate for the nominal system computed by letting in (10) and (11) $d = 0$ and $\delta = 0$. Then we find that $\|\hat{A} - \hat{A}^0\| \leq \|\hat{\Theta} - \hat{\Theta}^0\| \leq k_p(\bar{d} + \bar{d}^2 + \bar{d}^3 + \bar{\delta} + \bar{\delta}^2 + \bar{\delta}^3)$, which can be used directly to prove relation (17). For proving relation (18), let $\tilde{x} = x - \hat{x} = x - x^0 + x^0 - \hat{x}^0 + \hat{x}^0 - \hat{x}$ in the expression of $e_a(\tau)$ given by (14), where \hat{x}^0 is being generated by system (12) with \hat{A}^0 and \hat{b}^0 obtained from (10) and (11) with $d = 0$ and $\delta = 0$. By applying Lemma 7 to $x - x^0$ and $\hat{x}^0 - \hat{x}$, we obtain the result. \square

Let us consider now the transition factor given in expression (9) for system (6). Our aim is to find a possible value of the averaging time $\Delta\tau$ as function of noise level and parameter uncertainty such that for $\tau_i + \Delta\tau \leq \tau \leq \tau_{i+1} - \Delta\tau$ for each i the transition factor becomes smaller and smaller as the perturbation decreases and reaches zero when we have no perturbation at all.

Lemma 9 Let the transition factor be given by (9) for system (6). There exist positive constants c_1 and c_2 such that if

$$\Delta\tau = -c_1 \ln \left(\frac{1 - 2\beta}{1 - \beta} \right) \quad (19)$$

then the transition factor is such that

$$\text{Tr}(\tau) \leq c_2\beta, \quad \tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau, \quad (20)$$

$$\text{Tr}(\tau) \geq \frac{1 - \rho_0 - \varphi}{2}, \quad \tau = \tau_i, \quad (21)$$

for all i , where β and φ are perturbation dependent quantities and go to zero as the perturbation goes to zero.

Proof (sketch) . First note that each one of the systems in (6) can be written as $\dot{x} = Ax + b + w(t)$, where $w(t) = \delta Ux(t) + d(t)$, so that the expressions $\dot{x}_{av}(\tau^-)$ and $\dot{x}_{av}(\tau^+)$ can be computed by isolating the term deriving from $w(t)$. Then we compute $\gamma := \dot{x}_{av}(\tau^-)^T \dot{x}_{av}(\tau^+) / \|\dot{x}_{av}(\tau^-)\| \|\dot{x}_{av}(\tau^+)\|$, and by algebraic arguments we can derive that for $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$, $\gamma \geq [1 - k(1 - e^{\lambda\Delta\tau})](1 - \beta) - \beta$ for positive constants k and λ , where β depends on w ,

and it goes to zero as w goes to zero. In order to have $\gamma \geq 1 - (k + 2)\beta$, which leads to $\text{Tr}(\tau) \leq (k + 2)\beta$, we have $\Delta\tau = -1/\lambda \ln(1 - 2\beta/1 - \beta)$, which goes to zero when β goes to zero. For $\tau = \tau_i$ the proof proceeds similarly. \square

Proof of Theorem 4 . The proof proceeds in three steps: we show that the function W given in (15) achieves smaller values in $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$ than the one at $\tau = \tau_i$. Then we show that such value is larger also than the one that W achieves at times $\tau_i + \Delta\tau \leq \tau \leq \tau_{i+1} - \Delta\tau$ and, finally, at times $\tau > \tau_{i+1} - \Delta\tau$. Let us first show that $W(\tau_i) > W(\tau)$ for $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$. In fact

$$W(\tau_i) = \frac{\exp(\frac{-e_p(\tau_i)^2}{\sigma^2})\text{Tr}(\tau_i)}{a + e_a(\tau_i)} \geq \frac{\exp(\frac{-\Delta^2}{\sigma^2})(\frac{1-\rho_0-\varphi}{2})}{a + \varepsilon} \quad (22)$$

by virtue of Lemma 9, and Lemma 8 and by the fact that $e_a^0(\tau) = 0$ and $e_p^0(\tau) = 0$ for $\tau \leq \tau_i$. Also we have

$$W(\tau) = \frac{\exp(\frac{-e_p(\tau)^2}{\sigma^2})\text{Tr}(\tau)}{a + e_a(\tau)} \leq \frac{\text{Tr}(\tau)}{a} \leq \frac{c_2\beta}{a}$$

by virtue of Lemma 9, and Lemma 8 again. Therefore in order to show that $W(\tau_i) > W(\tau)$, it suffices to show

$$\frac{\exp(\frac{-\Delta^2}{\sigma^2})(1 - \rho_0 - \varphi)/2}{a + \varepsilon} > \frac{c_2\beta}{a},$$

which is the same as requiring

$$\frac{2c_2\beta}{a} \exp(\frac{\Delta^2}{\sigma^2})(a + \varepsilon) + \varphi \leq 1 - \rho_0. \quad (23)$$

Inequality (23) imposes conditions on the perturbation amplitude once ρ_0 has been fixed. In fact in the perturbed case if the noise and parameter uncertainties are too big even if in the nominal case the local signal properties would clearly give evidence of a transition, the corrupted signal could not maintain such local properties that would be hidden by perturbation.

Consider now the case $\tau_i + \Delta\tau \leq \tau \leq \tau_{i+1} - \Delta\tau$. For such times, from Lemma 5 and Lemma 6, we have $e_a^0(\tau) \geq k(\tau - \tau_i)^2$, then plugging this relation into the left side of (18) we obtain $e_a(\tau) \geq k(\tau - \tau_i)^2 - \varepsilon$. Using this relation in the expression of $W(\tau)$ we have

$$W(\tau) \leq \frac{\text{Tr}(\tau)}{a + k(\tau - \tau_i)^2 - \varepsilon} \leq \frac{c_2\beta}{a + k(\tau - \tau_i)^2 - \varepsilon}$$

where we have used (20). Since also inequality (22) holds, we require

$$\frac{\exp(\frac{-\Delta^2}{\sigma^2})(1 - \rho_0 - \varphi)/2}{a + \varepsilon} > \frac{c_2\beta}{a + k(\tau - \tau_i)^2 - \varepsilon},$$

and we find that $(\tau - \tau_i)^2$ should be bigger than

$$\max\left\{\Delta\tau^2, \frac{\varepsilon - a}{k} + \frac{2c_2\beta(\varepsilon + a)e^{\Delta^2/2}}{k(1 - \rho_0 - \varphi)}\right\} := (\Delta\tau^+)^2. \quad (24)$$

For such times $W(\tau)$ cannot have a maximizer, therefore the maximizer can occur only for $(\tau - \tau_i) \leq \Delta\tau^+$ which tends to zero as the perturbation tends to zero because when $\varepsilon \rightarrow 0$ and $\delta \rightarrow 0$, also $\beta \rightarrow 0$ and $\Delta\tau \rightarrow 0$ by Lemma 9, so that $(\Delta\tau^+)^2 = \max\{0, -a/k\} = 0$. In presence of perturbation, $(\Delta\tau^+)$ gives a measure of the uncertainty on τ_i for $\tau > \tau_i$, the left uncertainty is determined by $\Delta\tau$ only.

We finally show that $W(\tau_i) > W(\tau)$ for $\tau \geq \tau_{i+1} - \Delta\tau$. The approximation error $e_a^0(\tau)$ for the nominal system (7) is obtained by expression (14), that is $e_a^0(\tau) = \frac{1}{\tau - t_0} \int_{t_0}^{\tau} (\tilde{x})^T (\tilde{x}) dt$, where $\tilde{x} = x - \hat{x}$, and \hat{x} is generated by system (12), and $(\hat{A}|\hat{b})$ is obtained by expressions (10) and (11) with $d(t) = 0$ and $\delta = 0$. Then we can rewrite e_a^0 as $e_a^0(\tau) = \frac{1}{\tau - t_0} \left(\int_{t_0}^{\tau_i} (\tilde{x})^T (\tilde{x}) dt + \dots + \int_{\tau_{i+m}}^{\tau} (\tilde{x})^T (\tilde{x}) dt \right)$, and by applying Lemma 6 to each integral we have for a suitable h , $e_a^0(\tau) \geq h(\|(A_i|b_i) - (\hat{A}|\hat{b})\|^2 + \|(A_{i+1}|b_{i+1}) - (\hat{A}|\hat{b})\|^2 + \dots + \|(A_{i+m}|b_{i+m}) - (\hat{A}|\hat{b})\|^2)$. The term in parenthesis has a minimum for $(\hat{A}|\hat{b}) = (A^*|b^*)_{m+1} := ((A_i|b_i) + \dots + (A_{i+m}|b_{i+m}))/ (m+1)$ which is the barycenter of the distribution of $m+1$ points $\{(A_i|b_i), \dots, (A_{i+m}|b_{i+m})\}$. Denoting by $(A^*|b^*)_2$ the barycenter of the points $\{(A_i|b_i), (A_{i+1}|b_{i+1})\}$, we have

$$\begin{aligned} & \|(A_i|b_i) - (A^*|b^*)_2\|^2 + \|(A_{i+1}|b_{i+1}) - (A^*|b^*)_2\|^2 \\ & \leq \|(A_i|b_i) - (\hat{A}|\hat{b})\|^2 + \dots + \|(A_{i+m}|b_{i+m}) - (\hat{A}|\hat{b})\|^2 \end{aligned}$$

for any $(\hat{A}|\hat{b})$. Since also $\|(A_i|b_i) - (A^*|b^*)_2\|^2 + \|(A_{i+1}|b_{i+1}) - (A^*|b^*)_2\|^2 = (\|(A_i|b_i) - (A_{i+1}|b_{i+1})\|^2)/2$, it follows that $e_a^0(\tau) \geq h\|(A_i|b_i) - (A_{i+1}|b_{i+1})\|^2/2$. Assuming that $\|(A_i|b_i) - (A_{i+1}|b_{i+1})\| \geq \Delta_{min}^c$ for all i , we have $e_a^0(\tau) \geq h(\Delta_{min}^c)^2/2$. Then using relation (18) of Lemma 7, we find

$$W(\tau) \leq \frac{1}{a + h\frac{(\Delta_{min}^c)^2}{2} - \varepsilon},$$

where in place of $\text{Tr}(\tau)$ and of $\exp(-e_p(\tau)^2/\sigma^2)$ we have substituted one that is the maximum possible value they can take. Then since for $W(\tau_i)$ inequality (22) holds, it is sufficient that

$$\frac{1}{a + h\frac{(\Delta_{min}^c)^2}{2} - \varepsilon} < \frac{\exp(\frac{-\Delta^2}{\sigma^2})(1 - \rho_0 - \varphi)/2}{a + \varepsilon},$$

which implies

$$(\Delta_{min}^c)^2 > \frac{4e^{\Delta^2/\sigma^2}(\varepsilon + a)}{(1 - \rho_0 - \varphi)h} + \frac{2(\varepsilon - a)}{h} \quad (25)$$

This requirement asks that the minimum distance between parameters generating adjacent segments in the sequence of dynamical systems (6) has to increase if the perturbation due to parameter uncertainty and noise increases. In fact when τ increases after τ_i and multiple segments are included in interval (t_0, τ) the approximation error increases with respect to the one we have at $\tau = \tau_i$ where the only contribution is due to noise. If such increase is comparable with the contribution of noise that we have at $\tau = \tau_i$, then it becomes harder to say if we are including new segments in (t_0, τ) when τ increases. A way to prevent this is therefore to ask that the contribution to $e_a(\tau)$ when τ increases and new segments are included in (t_0, τ) is bigger than the one due to noise. This is guaranteed by a sufficiently big distance between $(A_i|b_i)$ and $(A_{i+1}|b_{i+1})$ for each i as expression (25) states. Also it is possible to show that constant k in expression (24) is proportional to (Δ_{min}^c) which means that for higher values of separations between points $(A_i|b_i)$ and $(A_{i+1}|b_{i+1})$ we are able to include a smaller portion of the segment starting at τ_i before realizing that a switch has occurred.

What we have shown up to this point is that $W(\tau)$ for $\tau \in [t_0 + \Delta\tau, \tau_l]$ has the global maximizer falling into $I = [\tau_i - \Delta\tau, \tau_i + \Delta\tau^+]$ if the noise level and parameter uncertainty are, for a given value of ρ_0 and (Δ_{min}^c) , such that conditions (23) and (25) are verified and if we take for $\Delta\tau$ the value specified in (19). To have the same result hold for $\tau \in (t_0, \tau_l]$ we need to assume that there is no switching point in $(t_0, t_0 + \Delta\tau)$. Therefore let $T = \min_i(\tau_i - \tau_{i-1})$ denote the shortest duration of a segment, and ask $\Delta\tau = -c_1 \ln\left(\frac{1-2\beta}{1-\beta}\right) < \frac{T}{2}$, which leads to

$$\beta < \frac{1 - e^{-T/(2c_1)}}{2 - e^{T/(2c_1)}}. \quad (26)$$

To complete the proof we need to show that the estimated class \hat{j} of the i^{th} segment is j . Denote with $\hat{\tau}_i$ the estimation of segmentation time τ_i . We have shown that $\hat{\tau}_i \in I$. If $\hat{\tau}_i \leq \tau_i$, we can apply Lemma 3 to obtain the classification result. Then consider the case $\hat{\tau}_i > \tau_i$. Let us denote by $(\hat{A}|\hat{b})$ the parameter estimate of $(A_i + \delta U_i|b_i)$, which can be computed by using equations (10) and (11). We are interested in considering the

distance of such estimate from $(A_i|b_i)$. Therefore:

$$\begin{aligned} (\hat{A}|\hat{b}) - (A_i|b_i) &= \left[\int_{t_0}^{\tau_i} (\delta U_i|0) \bar{x} \bar{x}^T dt + \int_{\tau_i}^{\hat{\tau}_i} ((A_{i+1}|b_{i+1}) - (A_i|b_i)) \bar{x} \bar{x}^T dt + \int_{t_0}^{\hat{\tau}_i} d(t) \bar{x}^T dt \right. \\ &\quad \left. + \int_{\tau_i}^{\hat{\tau}_i} (\delta U_{i+1}|0) \bar{x}^T dt \right] \left[\int_{t_0}^{\hat{\tau}_i} \bar{x} \bar{x}^T dt \right]^{-1}, \end{aligned}$$

and since $\hat{\tau}_i > \tau_i$ we have $\|\hat{A} - A_i\| \leq \|(\hat{A}|\hat{b}) - (A_i|b_i)\| \leq k_1 \bar{d} + k_2 \bar{\delta} + (\hat{\tau}_i - \tau_i) k_3 \leq k_1 \bar{d} + k_2 \bar{\delta} + (\Delta\tau^+) k_3$ for suitable positive constants k_1, k_2, k_3 . Since $A_i = A_c^j$ by Assumption 3.1, we want to obtain $\|\hat{A} - A_c^j\| \leq r_j$. Then it is sufficient that

$$k_1 \bar{d} + k_2 \bar{\delta} + (\Delta\tau^+) k_3 \leq r_j \quad (27)$$

where $\Delta\tau^+$ is defined in (24), so that the function $C(\hat{\tau}_i, k) := \|\hat{A} - A_c^k\| - r_k$ is less or equal than zero if and only if $k = j$, and therefore we have found $A_c^j = A_i$. Therefore by (23), (25), (26), (27) and Lemma 9, and recalling that $\varphi = \rho_0 \alpha + \alpha + \rho_0 \beta + \rho_0 \alpha \beta + \alpha \beta + \alpha$ and that $\beta = \beta(\bar{d}, \bar{\delta})$ and that $\alpha = \alpha(\beta) = \alpha(\bar{d}, \bar{\delta})$ we can derive conditions on the maximum allowed values for \bar{d} and $\bar{\delta}$. Let d^* and δ^* be such bounds. \square

Theorem 4 has been proved assuming $W(\tau)$ to be defined on (t_0, t_M) where $t_0 = \tau_{i-1}$ and $t_M = \tau_l$ is the duration of the process (6). The assumption that $t_0 = \tau_{i-1}$ is valid only at the first iteration in which $t_0 = \tau_0$. Then we find the maximizer $\hat{\tau}_1$ of $W(\tau)$ for $\tau \in (\tau_0, \tau_l)$ which lies in an interval $I = [\tau_1 - \Delta\tau, \tau_1 + \Delta\tau^+]$ around τ_1 and is an estimate of the first switching time τ_1 . Then we have to set t_0 for the second iteration so that the first switching point encountered after t_0 is τ_2 . In order to do this we set $t_0 = \hat{\tau}_1 + \Delta\tau$ so that we make sure that the first switching time encountered is τ_2 and not τ_1 again. In fact if the maximization process of W takes place with $t_0 > \tau_i$ and in the worst case scenario with $t_0 = \tau_i + \Delta\tau + \Delta\tau^+$ nothing changes as long as $T - (\Delta\tau + \Delta\tau^+) > 2\Delta\tau$ which by (19) and (24) implies an other condition on the noise level, which added to the ones found in Theorem 4 give new values for d^* and δ^* .

Remark 10 *If in the expression of the interconnection condition defined by (8) instead of \dot{x} we have just some of the components of the state vector, that is $\bar{C}\dot{x}$, and the same for the definition of the transition factor (9), provided that $\bar{C}\dot{x}_{av}^0(\tau^-)$ and $\bar{C}\dot{x}_{av}^0(\tau^+)$ are non zero, everything still applies.*

Remark 11 *Assume that in the expression of $W(\tau)$ given in (15) we add a factor $s(\tau)$ with the properties that $s(\tau) \in [\frac{1}{K}, 1)$ for all τ , $K \geq 1$ and $s(\tau) \geq 1 - \nu$ for*

$\tau_{i-1} < \tau \leq \tau_i$, with $\nu \ll 1$. Then the proof of Theorem 4 proceeds at the same way with minor modifications. Such a factor can depend on the classification of the current segment, and it can be introduced for some of the classes only. A good choice of such a term will be presented in the experiments section.

4.2 Algorithm implementation

The segmentation and classification algorithm was implemented in MATLAB 6.0 in the case of planar motion modeled in discrete time by systems of the form

$$X_{k+1} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ a_{1x} & a_{2x} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{1y} & a_{2y} \end{pmatrix} X_k + \begin{pmatrix} 0 \\ b_x \\ 0 \\ b_y \end{pmatrix} + d_k, \quad (28)$$

where $X_k = (x_{k-1}, x_k, y_{k-1}, y_k)^T$, with asymptotically stable dynamics in each interval between two switching points τ_{i-1} and τ_i . The interconnection condition that holds in this case is by replacing \dot{x} with $\bar{C}(x, \dot{x}, y, \dot{y})^T$ in equation (8), with $\bar{C} = ((0, 1, 0, 0)^T, (0, 0, 0, 1)^T)^T$. By virtue of Remark 10, the modified interconnection condition does not affect result of Lemma 9 since $\dot{x}_{av}^0(\tau^+)$ and $\dot{x}_{av}^0(\tau^-)$ and $\dot{y}_{av}^0(\tau^+)$ and $\dot{y}_{av}^0(\tau^-)$ are nonzero (since the dynamics are asymptotically stable.)

In the actual segmentation algorithm, $W(\tau)$ takes the following form

$$W(\tau) = e^{\left(-\frac{(e_a(\tau) - e_a^c)^2}{\sigma_a^2}\right)} \frac{\text{Tr}(\tau) e^{(-\bar{e}_p^2(\tau))} s(\tau) p(\tau)}{a + e_a(\tau)} \quad (29)$$

where we have introduced some slight modifications that exploit some additional information on the characteristics of the signal. In particular, the first factor represents a Gaussian distribution of the approximation error around a mean value: we can obtain a guess of e_a^c and σ_a^2 by processing part of the data. The parametric error \bar{e}_p takes into account also possible non-spherical shapes of the distribution of the parameters around the centers. Using the same notation used for defining e_p it can be written as $\bar{e}_p^2(\tau) = \min_j (\hat{A} - A_c^j)^T \Sigma_j^{-1} (\hat{A} - A_c^j) - \ln(1/\sqrt{\det(\Sigma_j)})$.

The term $s(\tau)$ is a shaping term, satisfying the properties described in Remark 11, which can be used to include additional information other than that derived from the dynamical parameters. Introducing such a term does not affect results of Theorem 4 as explained in Remark 11. The particular form we choose is introduced in Section 5.4.

The term $p(\tau)$ is introduced in case we have pauses in our signals. Pauses occur for the drawing tasks described in the next section and must be taken into account by the algorithm. Since we assumed that there are no pauses within a segment, eventual pauses are likely to be between one segment and the following one. If at time τ the segment (t_0, τ) contains a pause it will be penalized in an amount proportional to the length of the pause, i.e. $p(\tau) = k/(\text{pause length})$. Pauses are detected by making the difference between adjacent signal's samples and checking if the result is zero for more than 15–20 steps.

The time t_0 which is the starting point of each iteration is obtained as explained at the end of the proof of Theorem 4. The way we implement this is by taking into account that the end of each segment reaches a steady state in which \dot{x} and \dot{y} are very small and comparable to noise (since the systems are all stable asymptotically). Then we estimate the length of the signal after $\hat{\tau}_i$ which has a poor content of information with respect to a given threshold depending on the noise level. This gives an estimate of the time interval we have to add to $\hat{\tau}_i$ in order to find a point t_0 which lies in the following segment.

Finally, the segmentation and classifications algorithm can be described in the following basic steps, to be executed at each iteration: initialize $t_0 = \tau_0$, $t_M = \tau_l$; maximize $W(\tau)$ for $\tau \in (t_0, t_M]$ to get $\hat{\tau}_i$; compute class j of the segment found; compute $\Delta\tau$; set $t_0 = \hat{\tau}_i + \Delta\tau$ and $i = i + 1$; where we recall that τ_0 and τ_l are the starting and ending points of the data stream.

The theory so far developed does not address the case in which different movemes are represented by dynamical systems of different orders, which implies that different number of parameters describe different classes. In such a case the sets \mathcal{C}^j may lie in different spaces for different j . To account for this, in the definition of the parametric error (13) we can take the distance only between objects belonging to the same space, and since it is not known a priori what is the correct space for the class under study (because the class is not known), we should compute $\hat{A}(\tau, t_0)$ for all the different possible models and take the distance with the centers A_c^j lying in the same space. In this way the minimization in equation (13) is done over all possible j and over all possible model orders. This is the procedure that is used later on in the experiments, where the discrimination between a “reach” moveme and a “circle” moveme, which are represented by second order and fourth order models respectively, comes into play.

5 Experimental Results

To test our approach, we studied a 2D drawing task in which a set of shapes were drawn by five different subjects using a computer mouse (see examples in Figure 2).

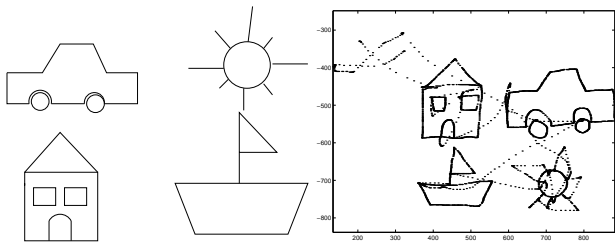


Fig. 2. Prototypes of the shapes (left) and example of traces in (x, y) captured by the capturing system (right.)

We hoped that these experiments could verify that human subjects use different dynamics for accomplishing different elementary tasks so to allow automatic recognition of actions. We considered just the motion of the wrist in 2D as a simple example to start with, and to develop a procedure for analyzing data easily generalizable and not subject to complexity constraints. In the following section we describe the device used for capturing data and details on the experiments.

5.1 Experimental Setup

Our subjects drew using the XPaint program on a PC running Red Hat Linux 7.2 with a screen measuring 1600×1200 pixels and a working window of 700×500 pixels. For acquiring x and y time traces we implemented a C routine which was activated in the background at the beginning of each experimental session and sampled the (x, y) position of the pointer everywhere on the screen at the rate of 100 Hz and a spatial resolution of one pixel. The routine makes use of XWindow libraries and captures the pointer position through the function “XQueryPointer” which is called by a timer every 10 ms and gives the coordinates in pixels with respect to the upper left corner of the screen. The data so obtained consists of an array with three columns containing time, x position at that time, y position at the same time. The time interval between one sample and the following one was mostly constant except for slight variations every once in a while due to higher priority of other processes. In order to have constant sampling time the data was processed through an algorithm that linearly interpolates data in the regions in which the time interval is not exactly 10 ms. Pixelization of the coordinates does not heavily affect the data since the trajectories under study are usually more than 50 pixels long.

We defined 4 different drawings by means of prototypes shown in Figure 2: car, sun, ship, and house. Each of the 5 subjects was shown the prototypes and was asked to reproduce them on a 700×500 pixel canvas; the only specification was to reproduce the prototypes with as high fidelity as possible in a reasonable amount of time. Each subject drew 10-20 examples for each shape.

In order to accomplish each drawing task the user had to perform a sequence of actions such as “reach a point

A” and “draw a line up to point B”. These actions are the ones that we will consider as candidates for being elementary motions and then defining a pair of movemes (for details see [5,6]). The idea is then to use the result of Theorem 4 so as to find the sequence of reach and draw movements that the user did in order to accomplish the task and the switching times between one and the other. In Figure 2 (right) we show also an example of the traces followed by a user while drawing in (x, y) plane.

5.2 Classification

We start from the hypothesis that “draws”, and “reaches” define a well-posed pair of movemes.

Before settling for model (28), we considered several other dynamical models for representing the reach and draw signals in time, starting from a first order, decoupled model for x and y motion

$$\begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = \begin{pmatrix} a_{1x} & 0 \\ 0 & a_{1y} \end{pmatrix} \begin{pmatrix} x_k \\ y_k \end{pmatrix} + \begin{pmatrix} b_x \\ b_y \end{pmatrix}, \quad (30)$$

and proceeding up to a second order coupled model

$$\begin{pmatrix} x_k \\ x_{k+1} \\ y_k \\ y_{k+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ a_{1x} & a_{2x} & a_{3x} & a_{4x} \\ 0 & 0 & 0 & 1 \\ a_{3y} & a_{4y} & a_{1y} & a_{2y} \end{pmatrix} \begin{pmatrix} x_{k-1} \\ x_k \\ y_{k-1} \\ y_k \end{pmatrix} + \begin{pmatrix} 0 \\ b_x \\ 0 \\ b_y \end{pmatrix}. \quad (31)$$

We segmented out by hand a set of straight draws from houses and cars drawn by 2 of the subjects. Reach examples were obtained from a special experiment session in which the users had to point and click at random buttons appearing on a 700×500 pixels window during a simple video game implemented in MATLAB 6.0. We considered 140 examples of reach trajectories and 140 examples of draw trajectories. The dynamical parameters were estimated for each one of the dynamical models proposed (first order for x and y , decoupled; first order for x and y , coupled; second order for x and y , decoupled; second order for x and y , coupled).

By proceeding with standard pattern recognition techniques (see [1] for example), we trained a Gaussian classifier for the parameters derived from the 140 examples per class (training set) for each one of the model classes proposed, and obtained the best results for the second order for x and y , decoupled, dynamical model (obtained by letting $a_{3x} = 0$, $a_{4x} = 0$, $a_{3y} = 0$, $a_{4y} = 0$ in system (31)), and reported in equation (28). For such a model we obtained 3.2% training error, and we tested the generalization properties of the resulting classifier on a test set of 323 additional reach examples (obtained from the

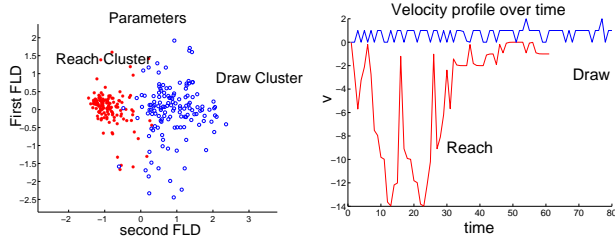


Fig. 3. Parameter estimates for reach and draw examples projected on the first two Fisher linear discriminants (left.) Typical velocity profile for reach and draw (right.)

MATLAB videogame) and 118 additional draw examples obtained from the drawings of other two subjects (different from the ones used for the training set) and obtained 3.63% test error.

Figure 3 represents the projection of the parameters belonging to the training set (living in \mathbb{R}^4) on the first two Fisher linear discriminants [1] and typical velocity profiles for the draw and reach trajectories. Those sets represent estimates of the sets \mathcal{C}^R and \mathcal{C}^D defined in Section 2, and we will refer to them as $\hat{\mathcal{C}}^R$ and $\hat{\mathcal{C}}^D$. From the right figure of Figure 3 we notice that a reach trajectory is usually characterized by a bell shaped velocity profile with high velocity variation in a small time, while a draw trajectory is characterized by an almost constant or slowly varying velocity.

Since our data set contains also circular shapes like the wheels of the cars, we also introduced a circle class beyond the reach and draw classes. The dynamical model by which we represent such a class is system (31), so that we have 8 parameters for classification. We considered an additional parameter that is the value of ω/T where ω is the principal frequency estimated and T is the duration of the trajectory: we expect for a circle that to be about 2π . We then trained a Gaussian classifier in \mathbb{R}^9 on a training set composed of 101 examples derived from the wheels of the cars and the suns of two of the subjects and obtained 4% training error on the circle class. We then tested the classifier on a test set of 124 elements derived from the wheels of the cars and suns of two other subjects (different from the one used for training) and obtained 8% error on the circle class. The higher test error on the circle class is due to higher variance across subjects than the variance we have for reach and draw tasks. The cumulative training and test errors for the tree classes are 3.4% and 4.6% respectively.

5.3 Segmentation algorithm performance

We implemented the proposed segmentation and classification algorithm in MATLAB on the data acquired as described in the previous sections considering a number of three movemes: the reach, the draw, and the circle movemes. Before reporting the algorithm performance

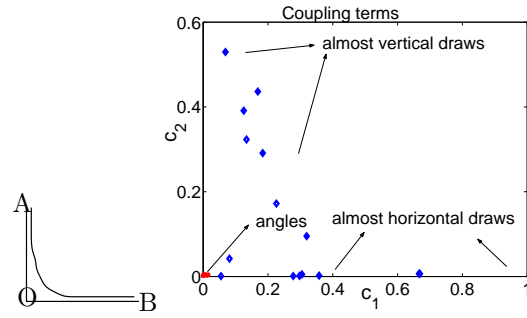


Fig. 4. The right plot reports the coupling parameters (stars) obtained for angles AOB shown in the left plot, and the coupling parameters (diamonds) obtained for the vertical and horizontal draws.

we describe the choice of the function $s(\tau)$ introduced in Section 4.2 in expression (29). The need for introducing a term with additional informations comes from the fact that system (28), chosen for representing the movemes, can approximate with acceptable approximation errors angles in x, y plane, as shown in the left plot of Figure 4, while having parameters that still lie in the $\hat{\mathcal{C}}^R$ or $\hat{\mathcal{C}}^D$ sets. Then the estimated parameters of system (28) for a given trajectory do not contain information to discriminate between one draw and an angle. This is due to the absence of any xy coupling information. In fact coupling information would discriminate quite clearly between the single draw case and the angle of Figure 4. We can check that for approximating an xy trajectory of the kind of the angle of Figure 4 with the simplest system containing xy coupling, such as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} d_1 & c_1 \\ c_2 & d_2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + b, \quad (32)$$

we obtain estimated coupling terms $(\hat{c}_1, \hat{c}_2)^T = \hat{c}$ that are close to zero for the angle and bounded away from zero for the single draw, as shown in the right plot of Figure 4 (the same we obtain for the single reaches that are approximatively straight lines.) Thus we choose a shaping term in expression (29), for the reach and the draw classes, of the form

$$s(\tau) = \frac{1}{1 + L \exp(-(\hat{c} - \bar{c})^T \Sigma_c^{-1} (\hat{c} - \bar{c})) / \sqrt{\det(\Sigma_c)}},$$

with $L \geq 1$, where \bar{c} and Σ_c are obtained by means of a learning phase in which we train the Gaussian classifier, $\exp(-(z - \bar{c})^T \Sigma_c^{-1} (z - \bar{c})) / \sqrt{\det(\Sigma_c)}$, on a set of about 25 examples of angles. The value of Σ_c turns out to be very small resulting in a very narrow Gaussian around the mean as we can deduce from the concentrated cluster of angle's parameters of Figure 4. By simple computation we can show that $s(\tau)$ satisfies the conditions of Remark 11.

Since in our data set some squares (windows of the houses) have rounded angles and look very similar to circles, we introduced a higher level step in the algorithm, in which we decide if a segment detected as a circle is more likely to be a square. At each iteration in which a circle is detected, to decide if the data segmented as a circle is more likely to be a square, we run the segmentation algorithm again on that data without the circle classifier (that is by assuming that the data is a sequence of reaches or draws or both). Then if the algorithm segments it into a sequence of draws, we compute the likelihood of each draw that has been detected as the product $\exp\left(-\frac{(e_a(\tau)-e_a^c)^2}{\sigma_a^2}\right) \exp(-\bar{e}_p^2(\tau))$, which is the part of (29) that quantifies how good the detected segment is as representative of its class. We then average the likelihood of all the detected draws and compare it to a threshold obtained by preprocessing some of the squares and some of the circles (about 10 examples each). This higher level process does not affect performance drastically, but turns out to be helpful in 3–4 cases in which the windows of the houses have not evident corners. Finally, for minimizing the algorithmic time, we set t_M *a priori* to be t_0 plus the maximum duration of a segment that in our case turned out to be 500 time steps.

The algorithm takes as input the signal $(x(t), y(t))$ and gives as outputs a sequence of segmentation points and the classification of the trajectory between each two detected segmentation points. The algorithm performance was computed by assuming a ground truth: we expected to detect a segmentation point at the beginning and at the end of each reach, draw and circle, and also we expected each one of them to be properly classified. Then the algorithm error was computed as the sum of classification error (i.e., percentage of trajectories that are correctly segmented but wrongly classified) and segmentation error (i.e., percentage of missed segmentation points and wrongly detected segmentation points). An estimate of such an error was computed on a data set of cars, ships, houses sequences deriving from two subjects each. The average error is about 10.5%. In Table 1 we show

Table 1
Algorithm error

	classification	segmentation	cumulative
CAR	$\frac{112}{1333} = 8.4\%$	$\frac{20}{1333} = 1.5\%$	9.9%
HOUSE	$\frac{108}{1050} = 10.29\%$	$\frac{23}{1050} = 2.19\%$	12.48%
SHIP	$\frac{99}{1093} = 9.06\%$	$\frac{3}{1093} = 0.27\%$	9.3%

the classification and segmentation errors separately for each one of the drawings considered. We report several pictures which show the segments classified as reach, the

segments classified as draws, the ones classified as circles and the unclassified ones. The little circles represent the segmentation points that the algorithm found. The units on the x and y axis are in pixels. From Table 1 it is evident that the major contribution to the algorithm error is due to classification error, in particular to confusion between reach and draw (see for example Figure 9) since it can happen that a subject draws quickly without paying enough attention so that some of the draws turn out to belong to another class of motions, called *free motion* that is not well-posed with respect to the draw and reach movemes (see [6,5] for details.) Also some of the reaches may be confused with draws because they are accomplished too slowly and carefully (especially when they are short). These errors are more

Table 2
Confusion matrix

	Predicted			
Actual	Reach	Draw	Circle	unclassified
Reach	167	32	0	10
Draw	35	395	0	12
Circle	0	0	44	2

clearly reported in Table 2 where we show the confusion matrix obtained counting the number of reaches, draws and circles in a set of cars and ships coming from one of the subjects. Some confusion is also due to the fact that some of the segments are too short, as for example in some of the windows of the houses, so that there is not enough information to classify them properly (see Figure 5.) This explains also why among the three different shapes (ship, car, house) the house is the one which shows the highest error. The segmentation error comes almost entirely from missed segmentation points and not from over-segmentation. The biggest portion comes from the windows of the houses in which some corners were not detected because of too small dimensions: at such small dimensions the hand dynamics is likely to vary with respect to the one used for larger motions and the pixelization error and the mouse dynamics may be not negligible anymore (see Figure 5.) Figure 6 shows houses for another subject. In Figure 9 we report for completeness the results of the algorithm on some of the suns: as we can notice by the figure, the classification error of the rays is about 50%. This is due to the fact that the draws of the rays have been shown to belong to the *free motion* class (see [5,6]). In Figures 7 and 8 we report results on cars and ships of subject 1.

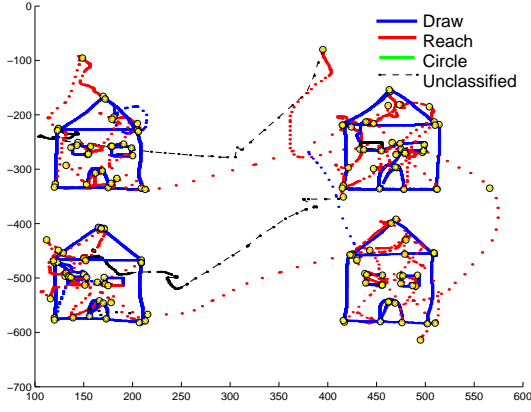


Fig. 5. Segmentation results on 4 houses of subject 4.

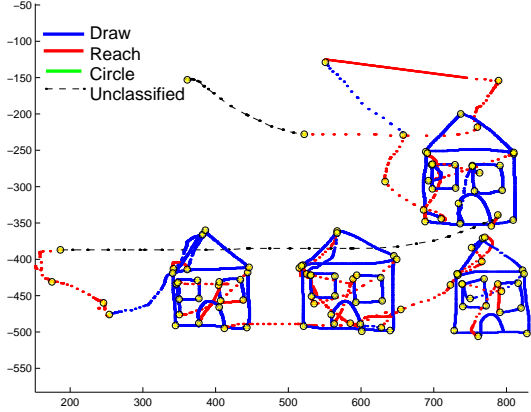


Fig. 6. Segmentation results on 4 houses of subject 2.

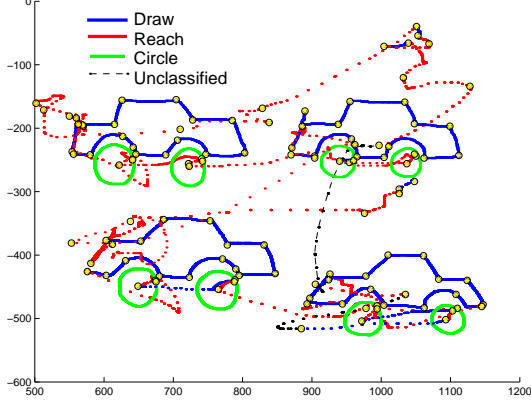


Fig. 7. Segmentation results on 4 cars of subject 1.

6 Conclusions

We have introduced a dynamical formulation for the notion of movemes. We have addressed the classification and segmentation problems and proposed an algorithm with error analysis. The experimental results show that the performance of the proposed algorithm is about 90% on our data set when training and testing are performed

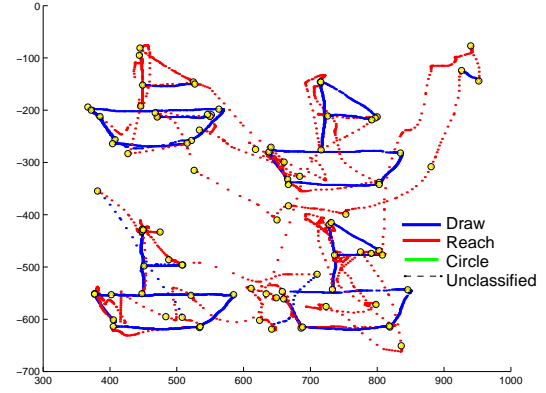


Fig. 8. Segmentation results on 4 ships of subject 1.

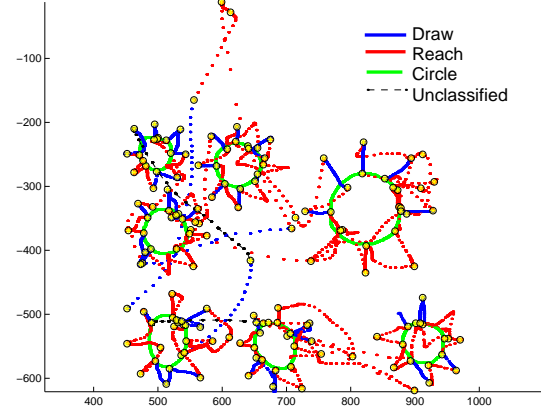


Fig. 9. Segmentation results on some suns of subject 1.

on data coming from distinct subjects. This gives evidence of the fact that the movemes considered are user-invariant on our data set. Subject-invariance is not a property that we can prove formally and requires an experimental verification. The results we obtain on 2D motions are encouraging in this respect.

The formalism that we introduced is directly applicable to the higher-dimensional case of full-body motion. If one compares it with previous work (e.g. the linear/quadratic input-output maps of [8]) one notices that our causal dynamical systems approach requires far fewer parameters for describing a moveme; hence it promises to require fewer training examples and allow for better generalization. Challenges into extending our results to three dimensional (3D) motion, which the current paper does not address, include the scalability of the approach, how to segment involuntary actions, and how to link moveme chains into meaningful activities. Additional work is also required to address issues like dependency on the number of training examples, and user-dependence of the movemes in a more complex and three dimensional experimental setting.

Furthermore, it is interesting to generalize the current segmentation and classification algorithm to the on-line

case. In the on-line setting it would be useful to think to a possible solution to the prediction problem, which is one of predicting the next action (or actions) on the basis of what has already happened. Moreover exploring different classes of dynamical systems may help modeling human motion with greater accuracy. Also issues regarding to what extent models are user independent and to what extent we need to train on different individuals should be addressed.

At an higher level of abstraction the idea of finding a “language” in which to specify what is possible and what is not seems to be promising. For example we know that in the sequence “step-step-reach-lift” for answering the phone, it is not possible to lift the phone before having reached it. These kinds of conditions could determine a model which gives a structure to the way in which movemes can be composed. A clear advantage of having such a model is that it could give feedback to the segmentation and classification algorithm so to increase its robustness.

Acknowledgements

This project has been funded in part by the NSF Engineering Research Center for Neuromorphic Systems Engineering (CNSE) at Caltech (NSF9402726), and by the ONR grant N00014-01-1-0890 under the MURI program. The authors would like to thank all of the people who participated to the experiments presented here.

References

- [1] C.M. Bishop. *Neural Networks for Pattern Recognition*. Clarendon, Oxford, 1995.
- [2] E. Bizzi and F.A. Mussa-Ivaldi. Toward a neurobiology of coordinate transformations. *New Cog. Neuroscience*, MIT Press, Cambridge, MA:489–500, 1999.
- [3] C. Bregler and J. Malik. Learning and recognizing human dynamics in video sequences. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 568–674, Puerto Rico, 1997.
- [4] R.T. Collins, A. J. Lipton, and T. Kanade. Introduction to the special section on video surveillance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22:745–746, August 2000.
- [5] D. DelVecchio, R. M. Murray, and P. Perona. Decomposition of human motion into dynamic based primitives with application to drawing tasks. Technical report, Caltech, report number 2002-004, URL: <http://www.cds.caltech.edu/murray/papers/index.shtml>, aug 2002.
- [6] D. DelVecchio, R. M. Murray, and P. Perona. Classification of human actions into dynamics based primitives with application to drawing tasks. In *European Control Conference*, University of Cambridge, UK, 2003.
- [7] J.R. Flanagan and A.M. Wing. The role of internal models in motion planning and control: evidence from grip force adjustments during movements of hand-held loads. *The Journal of Neuroscience*, 17:1519–1528, 1997.
- [8] L. Goncalves, E. Di Bernardo, and P. Perona. Reach out and touch space (motion learning). In *Proc. of the Third International Conference on Automatic Face and Gesture Recognition*, pages 234–239, Nara, Japan, April 14-16 1998.
- [9] L. Goncalves, E. Di Bernardo, E. Ursella, and P. Perona. Monocular tracking for human arm in 3d. In *Proc. of the 7th Int. conference on Computer Vision, ICCV*, pages 764–770, 1995.
- [10] F. Gustafsson. *Adaptive Filtering and Change Detection*. John Wiley & Sons, 2000.
- [11] M. Kawato. Internal models for motor control and trajectory planning. *Current Opinion in Neurobiology*, 9:718–727, 1999.
- [12] I. Laptev and T. Lindeberg. Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features. In *IEEE Workshop on Scale-Space and Morphology*, pages 63–74, Vancouver, Canada, July 2001.
- [13] M. Lavielle. Optimal segmentation of random processes. *IEEE Trans. on Signal Processing*, 46:1365–1373, May 1998.
- [14] L. Ljung. *System Identification*. Prentice Hall, New Jersey, 1999.
- [15] C. Lu, H. Liu, and N.J. Ferrier. Multidimensional motion segmentation and identification. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 629–636, Hilton Head Island, South Carolina, 2000.
- [16] M.E. Munich and P. Perona. Visual input for pen-based computers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:313–328, March 2002.
- [17] F.A. Mussa-Ivaldi, S.F. Giszter, and E. Bizzi. Linear combinations of primitives in vertebrate motor control. *Proc. of the National Academy of Science*, 91:7534–7538, 1994.
- [18] D. Ormoneit, T. Hastie, and M.J. Black. Functional analysis of human motion data. In *Proc. 5th World Congress of the Bernoulli Society for Probability and Mathematical Statistics and 63rd Annual Meeting of the Institute of Mathematical Statistics*, Guanajuato, Mexico, 2000.
- [19] V. Pavlovic and James M. Rehg. Impact of dynamic model learning on classification of human motion. In *IEEE Conf. Computer Vision and Pattern Recognition*, Hilton Head Island, 2000.
- [20] T. Söderström and P. Stoica. *System Identification*. Prentice Hall. Hemel Hempstead, 1989.
- [21] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, 1995.
- [22] S. Waldherr, S. Thurn, R. Romero, and D. Margaritis. Template-based recognition of pose and motion gestures on a mobile robot. In *Proc. of the AAAI 15th National Conference on Artificial Intelligence*, pages 977–982, 1998.
- [23] P. Wellner. The digital desk calculator: Tactile manipulator on a desk top display. In *Proc. of the ACM Symposium on User Interface and Technology*, pages 27–33, Hilton Head, November 1991.
- [24] A.S. Willsky and H.L. Jones. A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Trans. on Automatic Control*, 21:108–112, February 1976.
- [25] A. Wilson and A. Bobick. Learning visual behavior for gestures analysis. In *Proc. of IEEE Symposium on Computer Vision*, pages 229–234, Coral Gables, FL, November 1995.
- [26] Y. Yacoob and L. Davis. Recognizing human facial expressions from long image sequences using optical flow. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 18(6), pages 636–642, 1996.